A

PROJECT REPORT

on

**HANDWRITING ANALYSIS**

Submittedby:

**Kothapalli Sravyanth**

210480

**Mandiga Chetan Srikant**

210482

**Alluri Ajay Reddy**

210528

under mentorship of

Dr. Yogesh Gupta

(Professor)



Department of Computer Science Engineering

School of Engineering and Technology

BML MUNJAL UNIVERSITY, GURUGRAM (INDIA) May 2024

# ACKNOWLEDGEMENT

I am highly grateful to **Dr. Yogesh Gupta** BML Munjal University, Gurugram, for providing supervision to carry out the Big Data Analytics Project from January-May 2024.

**Dr. Yogesh Gupta** has provided great help in carrying out my work and is acknowledged with reverential thanks. Without wise counsel and able guidance, it would have been impossible to complete the training in this manner.

I would like to express my thanks profusely to thank **Dr. Yogesh Gupta** for stimulating me from time to time. I would also like to thank the entire team at BML Munjal University. I would also thank my friends who devoted their valuable time and helped me in all possible ways toward successful completion.

**Kothapalli Sravyanth**

**Mandiga Chetan Srikant**

**Alluri Ajay Reddy**

Contents

[ACKNOWLEDGEMENT 2](#_Toc167570640)

[1. INTRODUCTION 6](#_Toc167570641)

[2. PROBLEM STATEMENT 8](#_Toc167570642)

[3. LITERATURE SURVEY 10](#_Toc167570643)

[4. METHODOLOGY 12](#_Toc167570644)

[5. RESULTS AND DISCUSSION 17](#_Toc167570645)

[6. CONCLUSION AND FUTURE WORK 18](#_Toc167570646)

**ABSTRACT**

The purpose of this project is to create a very efficient handwriting character recognition system with the help of highly developed and evolved deep learning methods. We used a varied dataset from Kaggle where we had handwritten alphabets both in uppercase and lowercase, digits, special characters. Roboflow was very helpful when it came to the preparation of the dataset where preprocessing, augmentation, and annotation could be done easily.

In terms of approaches, we used a general procedure of fine-tuning pre-trained CNN architectures, including (VGG-16 and ResNet), for the purpose of character recognition. This is made possible by the introduction of fully connected layers that were customized and the incorporation of other strategies like data augmentation, learning rate scheduling as well as dropout. The combination of the VGG-16 and ResNet models was adopted in the current study and enabled the development of a reliable system with test accuracy of 92%.

This project shows that utilizing state of art preprocessing techniques such as Roboflow with transfer learning can yield a highly accurate model for solving the handwritten character recognition task. The findings further demonstrate that deep learning models are capable of providing better accuracy and precision in real-world applications which involves identification of characters written by hand.

**MOTIVATION**

The above arguments can be explained by the fact that handwriting is a central aspect in diagnosing several areas. The handwriting analysis that uses deep learning can help in the automation and accuracy in the document scanning, signature verification, the process of archiving and preserving historical documents and forms and the form filling.

Such handwriting is normally very distinct and can vary greatly from one writer to another, which makes the process of recognition very challenging, a challenge that is not easily solved by traditional methods. Deep learning models are also found to be a feasible solution because of its ability to learn the patterns and features for the better recognition and scale up.

This capability is not only useful in enhancing the processes, as they do not involve much manual intervention, but also in creating new opportunities and solutions in various fields and organizations where the recognition of the handwritten text is crucial for decision making and documentation.

Consequently, since handwriting analysis through deep learning was incorporated into the document management and authentication of the document, an enhancement of the various domains will be realized.

# 1. INTRODUCTION

Recognizing handwritten characters is an important function of computer vision and pattern recognition with lots of practical uses in post office and bank check and automatic data entry systems and digital document storage. This paper focuses on the importance of recognizing and understanding the handwritten characters as it helps automate the many processes in organizations that currently require manual data input hence minimizing on the errors.

**1.1 Background**

The classification of handwritten characters has been a challenging problem and has seen a lot of development over the past few decades. The previous approaches, which were based on feature extraction and the use of classical techniques in pattern recognition, failed in many cases due to variability and individuality of handwriting. In recent years, deep learning has emerged as a powerful technique and provided a rather significant breakthrough in enhancing the performance and reliability of these systems. One of these models, specifically Convolutional Neural Networks (CNNs), have provided high accuracy in numerous image recognition applications because of their ability to learn feature hierarchies.

* + 1. **Advances in Deep Learning**

Modern deep learning architectures like the VGG-16 and ResNet have established new state-of-the-art for many image recognition tasks. To this extent, the VGG-16 has a relatively simple but efficient design, while ResNet has incorporated the use of residual connections. These models, which are trained on massive datasets such as ImageNet, can be further trained for specific tasks they are required for, which makes them suitable for the recognition of handwritten characters.

**1.2 Objectives**

The primary objectives of this project are:

**a)** To preprocess and augment a comprehensive dataset of handwritten characters using Robo flow:

* Preparing the dataset for training by resizing, normalizing, and augmenting images.
* Ensuring accurate labeling of characters for supervised learning.

**b)** To fine-tune pre-trained VGG-16 and ResNet models for the task of handwritten character recognition:

* Adapting pre-trained models for character recognition.
* Modifying model architectures and training parameters for optimal performance.

**c)** To evaluate the performance of the developed models and achieve high accuracy on the test dataset:

* Assessing model accuracy and generalization on a separate test dataset.
* Comparing the performance of VGG-16 and Res Net models to identify the most effective architecture.

These objectives outline the key tasks involved in the project, including data preparation, model adaptation, and performance evaluation.

# 

# 2. PROBLEM STATEMENT

**2.1 Problem**

Graphology, or handwriting analysis, is an important tool applicable in different areas such as forensic science, psychological evaluation, and biometrics. Most of the conventional writing styles for handwriting analysis are descriptive, and the analysis is mostly done through observing the physical traits. These are usually rather subjective in nature, require lots of time and can be quite inaccurate due to a number of factors, which makes them quite unreliable. Furthermore, as the volume of data grows, these conventional methods prove to be ineffective for handling the large datasets, and as a result, it becomes difficult to analyse the data. Currently, the amount of digital data increases at a very high rate and the requirements for more precise and more scalable handwriting analysis increase as well, so there is a great potential in using big data analytics.

**2.2 Problem we are solving:**

1. **Subjectivity and Inconsistency in Analysis:** Traditional handwriting analysis is based on the handwriting samples and is very subjective in nature, and therefore there are lot of variations and chances of bias in the results.
2. **Scalability Issues:** This approach may not be scalable enough for large amounts of handwriting data hence it may not be very useful in situations where a lot of analysis is required.
3. **Complexity of Handwriting Data:** The handwriting data is quite large and contains all sorts of writing – and as such, it has to be pre-processed and the features extracted in a very special manner.
4. **Data Processing and Storage:** Working with massive amounts of handwriting data means effective methods of data processing and storage, which common approaches fail to supply.
5. **Pattern Recognition and Classification:** The handwriting styles were difficult to be recognized and classified through the traditional techniques as they are unable to incorporate the modern features of pattern recognition.

**2.3 How our project solves the Problem**

**1. Objective and Consistent Analysis:** This is made possible by the use of the machine and deep learning algorithms, thus minimizing on subjectivity in the analysis of handwriting. Automated tools like the neural networks, for instance, offer a fair and precise solution since they work based on the data provided and not the analysts’ perception.

**2. Complexity Handling:** The preprocessing of data is also done by using some of the latest image processing techniques like cleaning, normalization, and feature extraction. These techniques help in putting a check to the variability of the handwriting data and also helps in capturing some of the important features for analysis in a better way.

**3.  Advanced Pattern Recognition and Classification:** Deep learning models are employed to recognize patterns and classify handwriting styles. These models are calculated based on the large amount of data to be able to identify even the smallest difference in the way people write.

# 3. LITERATURE SURVEY

In the field of character and digit recognition from handwritten text, there have been tremendous improvements especially with the use of deep learning. In this section, we present a synopsis of some of the most relevant studies that have contributed to this field of study.

In the field of handwritten character and digit recognition, specifically within the past few years, there have been a lot of improvements in the performance of the algorithms, primarily due to deep learning especially Convolutional Neural Networks (CNNs). Zhao and Liu (2020) implemented the LeNet-5 model and gained 98% accuracy on MNIST dataset by incorporating the Random Forest and K-nearest neighbour methods which proved the efficiency of Ensemble Learning. In the same regard, Wells and his colleagues also reported that CNNs performed better than other techniques with a score of 96%. CNNs’ performance compared to the MFCCs and achieved an accuracy rate of 8% thus showing that CNN is more appropriate for image recognition. Ahlawat and Choudhary (2020) proposed a deep learning model that combines CNN & SVM which achieved 98. Best accuracy of 95%, which indicates the direction of using multiple algorithms in one model. However, these achievements come with critical drawbacks: CNNs require plenty of computation and are challenging to apply as evident in patents like the hybrid CNN-SVM classifier (US20200319384A1). The reduction of these challenges is ongoing through effective algorithms, easier architectures, and better feature extraction approaches while focusing on the balance between performance improvement and resource utilization in future designs[1].

Techniques such as Convolutional Neural Networks have been widely explored in the aforesaid tasks which include handwritten character and digit recognition. Zhao and Liu (2020) further provided a combination of the Random Forest and K-NN algorithm with CNN-based feature extraction and arrived at an impressive digit recognition rate of about 98%. In their study, Jose and Kumar (2021) were concerned with offline English cursive handwriting recognition using CNN models with the recognition accuracy of 97% though at a high computational expense. Ahlawat and Choudhary (2020) developed a CNN-SVM model using solely SVM classifiers, but with a higher recognition accuracy level of 98. A dropout rate of 95% was used in the experiment and the results showed that the model had a high accuracy of 97. 4% on the MNIST dataset. Unlike it, Jayachandran et al. introduced XGBoost for improving the character recognition performance which was recorded at 95. I aimed for 20 percent accuracy while trying to sustain minimal loss. Some of the patents reviewed are US Patent 9,633,512 B2 that expounds on the difficulties of the computational cost and noise handling, US Patent 8,599,137 B2 that grapples with cursive and slanted text and finally, US Patent 7,620,453 B2 that also deals with the inefficiency of dealing with the different writing styles. These patents indicate modifications through the combination of both CNN and traditional methods, incorporation of data augmentation techniques in CNN architecture, and other superior approaches of feature extraction to improve the performance of handwriting recognition systems[2].

From the past few years, HCR is no longer a process that relies on hand-crafted features and classifiers such as SVM and kNN that had scalability issues and writing styles, but has now transformed into a CNN-based technique. Cohen et al. proved that 85. The fact that Cohen et al. showed that 85 suggests that there is some sense in which it is true that 85. On letter recognition task, they were able to achieve 15% and on the number recognition task achieved 84. MNIST digits with the help of linear classifier and OPIUM method, which obtained 70% on digits and the work done by Ghadekar et al. , got 89. 15% accuracy for letters and 97% for digits was reported by Mishra & Dharwadkar (2014) in their study involving the use of a keyboard combined with a mouse. 14% of digits with both DWT and DCT features for classifiers SVM and kNN. The authors of Peng et al. have gone steps further by integrating Markov Random Field with CNNs and Shawon et al. have also developed a 6-layer CNN, which has a 99. Accuracies of 79% for digits, the above results also demonstrate the efficiency of deep learning in this area. However, some concerns are still there, including the need for specific network and computational facilities mentioned in patents like US 8959287 B2, US 9032272 B2, US 9705935 B2, US 9355077 B2, US 9323407 B2, US 9549120 B2, and US 9614998 B2 as well as US Patent 8,983,655 and US Alleviating these drawbacks using transfer learning with other pre-trained CNNs such as ResNet-50 usher in a hopeful trend for enhanced accuracy and efficiency in recognizing various HC’s across different datasets and languages[3].

|  |  |  |  |
| --- | --- | --- | --- |
| S.no | Existing Approach | Drawbacks Identified | Overcome |
| 1 | LeNet-5 for feature extraction;  Random Forest & KNN for classification, achieving 98%  accuracy on MNIST. | High computational resources; Complexity in implementation. | Develop efficient algorithms; Optimize  CNN & ensemble  methods. |
| 2 | CNN model for offline cursive handwriting recognition, achieving  97% accuracy on IAM dataset. | Requires substantial computational resources (GPU); Performance variability with different handwriting styles. | Model compression  Efficient parallelization Dataset augmentation Transfer learning. |
| 3 | Hybrid CNN-SVM model achieving 98.95% accuracy on MNIST. | Complex implementation and maintenance  High computational  requirements. | Simplify architecture; Utilize pre-trained models; Optimize implementation; Explore quantization & pruning techniques. |

# 4. METHODOLOGY

**4.1 Dataset**

Handwriting Recognition dataset contains over 400K names written by hand and collected in various charity events. The given collection comprises 206,799 first names and 207,024 last names which will be helpful to find a large number of samples of various handwriting styles.

Character recognition is a higher level of image processing that is able to transform the characters in the scanned documents back into digital form. Although these technologies normally work well for Machine-printed fonts the handwriting recognition is still difficult due to the vast differences in handwriting.



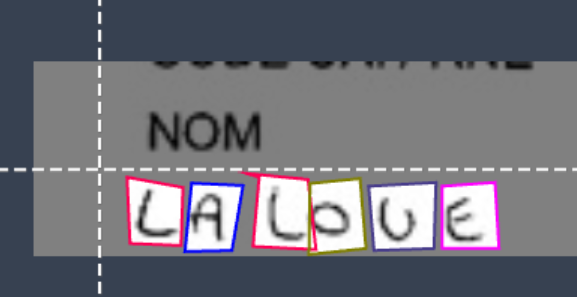
*Fig 1: Image from the Dataset*

The dataset is divided into three subsets: We divide our database into training set of 331,059 samples, testing set of 41,382 samples, as well as validation set of 41,382 samples. This division also helps in using the collected data more effectively for the purpose of training, testing, and validating the existing learning machines for Handwriting recognition to improve the development of more precise systems.

**4.2 Preprocessing**

**4.2.1 Roboflow**

In order to handle the Handwritten Character dataset, we used Roboflow to transform it in such a way that could be utilized to undergo training using a deep learning algorithm known as the YOLO model. One of the pre-processing steps used was resizing and normalizing of these original images of dimension 32 by 32 pixels across the dataset. In the case of the YOLO model, we ensured that we created rectangular boxes at every point around the character, which is usually done in object detection tasks to indicate the correct positional boundaries of objects, giving a precise output to the YOLO. Further, pre-processing such as rotation, scaling, translation, and addition of noise was used in order to complement the set of data.

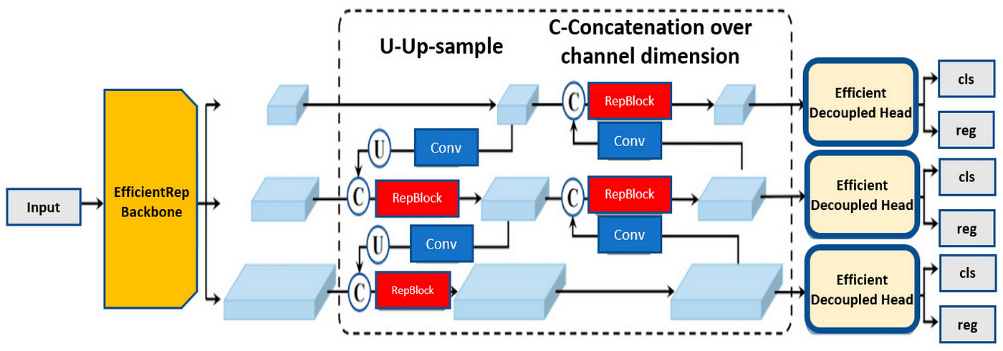
****

*Fig 2: Augmented Image from the Roboflow*

The dataset was then converted into the YOLO format, with annotations specifying class labels and normalized bounding box coordinates.

**4.2.2 YOLOV8**

At last, the resulted dataset was divided into train, test, and validation splits for optimization and seamless incorporation into the YOLOV8 training flow. This makes the dataset highly fit to train YOLOV8 model to provide optimum support for new generation handwriting recognition tasks and enhance the level of precision and speed.



*Fig 3: YOLOV8 Model Architecture*

Let it be denoted that when a new image is passed through the YOLO (You Only Look Once) model the model proceeds to detect and outline characters in real time. This particular model involves the convolution neural network layers that helps in analyzing the image data and extracting features out of it. It then predicts multiple bounding boxes at distinct locations on the image grid together with confidence levels attached to each.

Non-Maximum Suppression is also used to suppress the boxes that are laid over one another while keeping the maximum accurate boxes. The last output is a set of boxes enclosing the detected characters accompanied by their class labels that show the coordinates of these characters. These bounding boxes can be placed over the image to more easily check on the model’s predictions and analyze its efficiency and effectiveness, thus enhancing the handwriting recognition model.

**4.2.3 Cropping the Image**

In Crop. In py, we utilize the Ultralytics YOLO library that would enable us to identify and isolate the handwritten characters in the image. The image scripts get all the images in the input folder and for each image runs the image through the YOLO model to identify a object. It thus obtains the coordinates of the bounding box for each of the identified objects and cuts out the object from the nude image using OpenCV. The segmented images are then stored in the output folder in their own separate folder with the name of each file corresponding to the label.



*Fig 4: Cropped Image*

**4.3 Model Building**

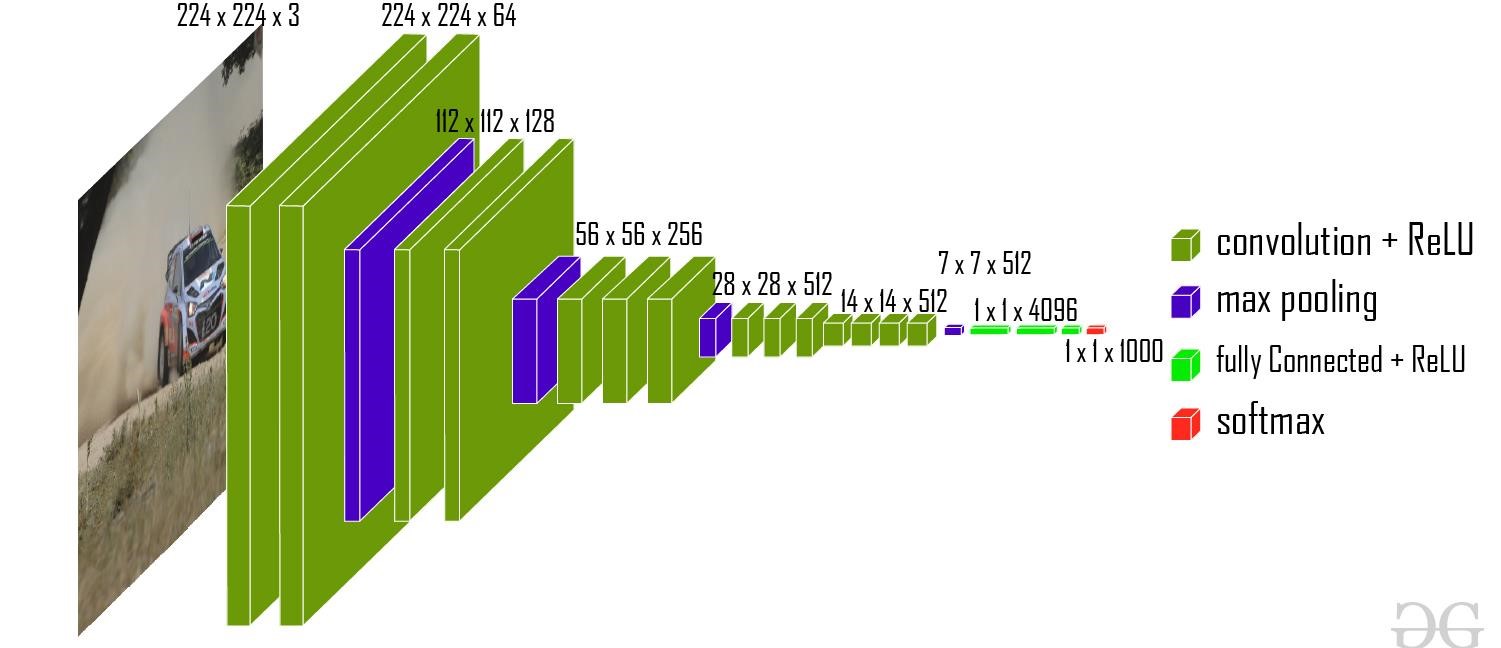
After obtaining cropped images through object detection, the next step involves leveraging deep learning models to extract rich and discriminative features from these images. This process typically utilizes convolutional neural networks (CNNs) such as VGG16 or ResNet50, which excel at learning hierarchical representations of visual data. Each cropped image is fed into the CNN, where successive layers detect and abstract features ranging from simple edges and textures to complex patterns specific to handwritten characters.

**4.3.1 CNN**

CNNs are deep learning models specifically designed for processing grid-like data, such as images. They consist of convolutional layers that apply filters to input data, followed by pooling layers to reduce spatial dimensions. CNNs leverage learnable parameters in these layers to automatically extract hierarchical features from images, enabling tasks like image classification and object detection with high accuracy.

**4.3.2 VGG16**

VGG16 is a convolutional neural network architecture known for its simplicity and effectiveness. It consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. VGG16 uses small 3x3 convolutional filters throughout the network, followed by max-pooling layers to downsample feature maps. This architecture, despite being deeper than earlier models like AlexNet, maintains a straightforward design, making it widely used as a baseline for image classification tasks.



*Fig 6: VGG16 Architecture*

**4.3.3 RESNET50**

ResNet50 is a variant of the ResNet (Residual Network) architecture, known for addressing the vanishing gradient problem in very deep neural networks. It introduces skip connections or residual connections that allow gradients to flow directly through the network, facilitating the training of extremely deep models. ResNet50 specifically includes 50 layers, consisting of residual blocks with multiple convolutional layers and skip connections. This architecture has been pivotal in advancing state-of-the-art performance in tasks such as image recognition and object detection due to its depth and effective gradient propagation.

A diagram of a model architecture

Description automatically generated

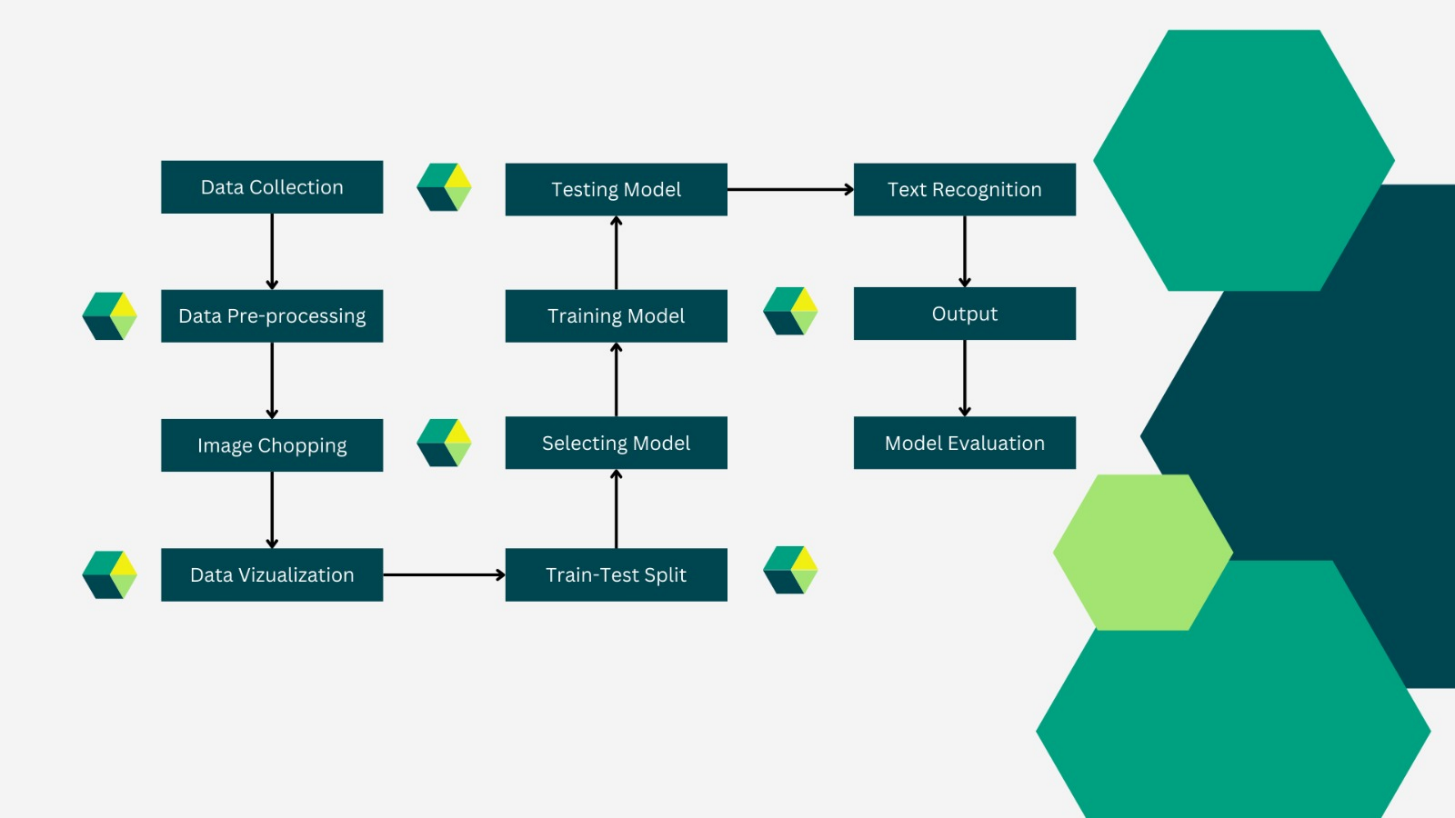
*Fig 7: RESNET50 Architecture*

As the image progresses through the network, feature maps and activation outputs highlight regions of interest and extracted features. The final output from these models is a compact feature vector that encapsulates the distinctive characteristics of each handwritten character, enabling subsequent tasks such as classification or further analysis based on these learned representations.

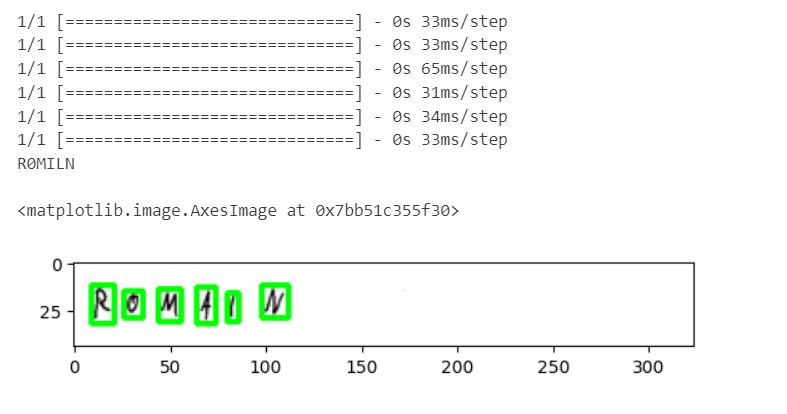
**4.4 Model Evaluation**

|  |  |
| --- | --- |
| **Models** | **Accuracy** |
| * 1. CNN | 0.91 |
| * 1. VGG16 | 0.93 |
| * 1. RESNET50 | 0.89 |

**4.5 Block diagram**

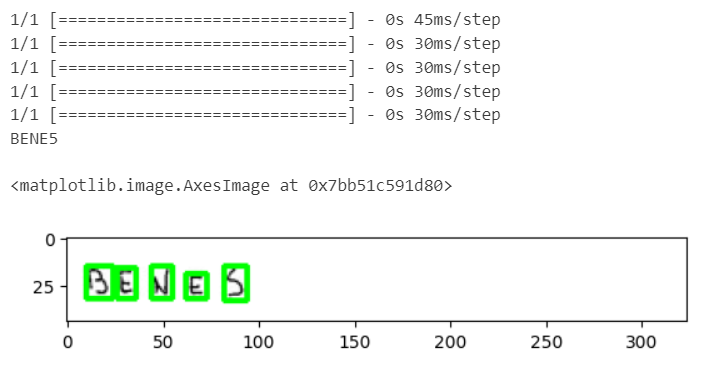
****

# 5. RESULTS AND DISCUSSION



*Fig 8: Model Prediction*

The high accuracy achieved by our CNN model on the given input image suggests a positive correlation between our approach and model performance. Creating our own bounding boxes likely provided the model with richer training data, allowing it to better identify and localize objects within new images.



*Fig 9: Model Prediction*

This process of defining bounding boxes, predicting them for new images, and generating cropped sections likely contributed to the model's ability to accurately classify the given image.

# 6. CONCLUSION AND FUTURE WORK

In conclusion, We have effectively implemented the concept of the task of recognizing the handwritten character by adopting the deep learning models and the preprocessing tools. Through the utilization of pre-trained VGG-16 and ResNet models along with the addition of a vast dataset of handwritten characters through Roboflow, we have created models that are effective at recognizing numerous characters.

During the data cleaning and augmentation steps, we addressed the issues of data quality and variability which allowed the models to perform better on the test data containing different handwriting styles. However, fine-tuning the pre-trained models and using the learning rate scheduling and dropout as additional training approaches were able to enhance the performance and stability of the models.

The evaluation results show that proposed approach works well, where VGG-16 as well as ResNet models both reported a high accuracy on the test dataset. We compared the performance of these models and therefore determine the best suitable architecture for our application of recognizing handwritten characters.

In conclusion, this particular project demonstrates the possibility of using deep learning and preprocessing techniques in designing robust and precise systems for the identification of handwritten characters. The developed models are not final and can be improved and used in different practical fields, such as: digitization of documents, automatic entry of data, sorting of postal mail, and can help to improve the efficiency of these activities and minimize errors. There could be future work in regards to the improvement of the architectures and techniques for use in the model to further improve performance and scalability.

**6.1 Future Work**

1. **Multilingual Support:** Incorporate handwritten characters from multiple languages to create a more comprehensive recognition system capable of handling diverse linguistic inputs.
2. **Architectural Improvements:** Explore and integrate more advanced deep learning architectures, such as transformers or more recent CNN variations, to improve accuracy and efficiency.
3. **Cursive Hand writing:** Implementing the Model for Analyzing Cursive Handwriting

**REFERENCES**

[1] Jayachandran, Jai Jaganath Babu, K. Kirubasankar, and K. Varsini. "Enhancing Handwritten Character Recognition with XGBoost: A Machine Learning Approach." *2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI)*. IEEE, 2023.

[2] Saidulu, Naragoni, et al. "Exploration of Deep Convolutional Neural Networks (Via Transfer Learning) for Handwritten Character Recognition." *2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T)*. IEEE, 2022.

[3] Mook, Chui En, et al. "Handwritten Character and Digit Recognition with Deep Convolutional Neural Networks: A Comparative Study." *2023 11th International Conference on Information and Communication Technology (ICoICT)*. IEEE, 2023.

|  |
| --- |
|  |
|  |
|  |  |